Promoter characterization

2D

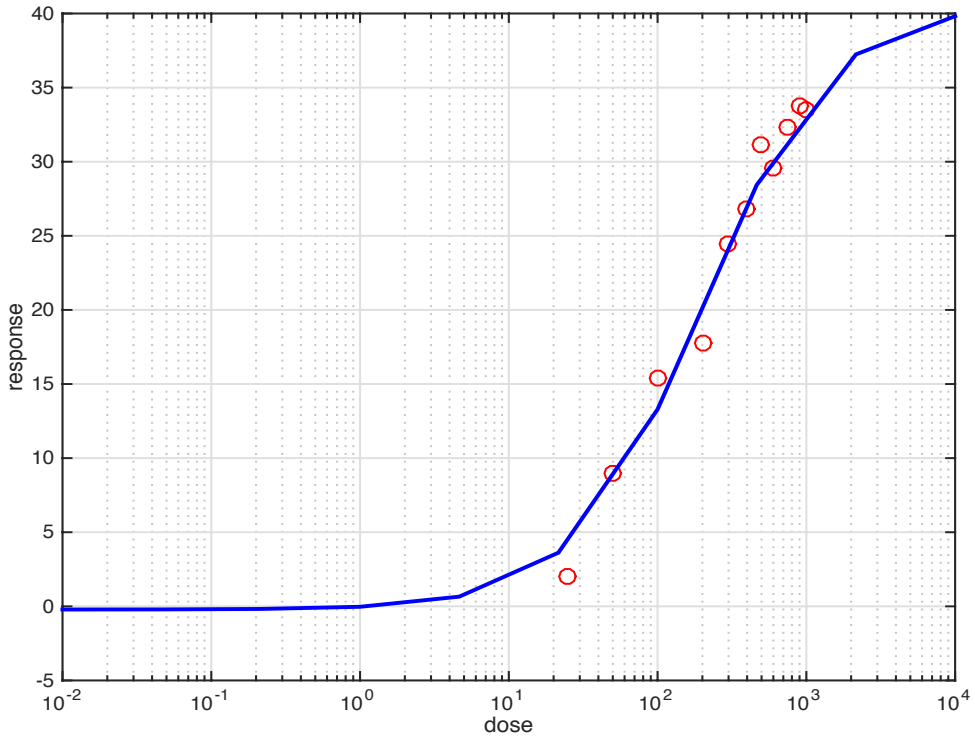

Figure 1. Non-Linear Fit of a Dose Response Curve. Matlab script calls function “hill.m”, which takes input values for x and y of a completed dose-response experiment and plots them. The script then runs statistical analysis of the data and compares with an external function 'hill.m' that contains the formula to fit the data to which is the hill equation function (for either repression or activation). When called, the hill equation function outputs the calculated values for the maximum value (ymax), dissociation constant (ka) and the hill coefficient (n). It is required to input approximate guesses of what the parameters might be to help with the regression. Outputs a graph with a fitted curve and the parameter values that best match the data are plotted above.

3D

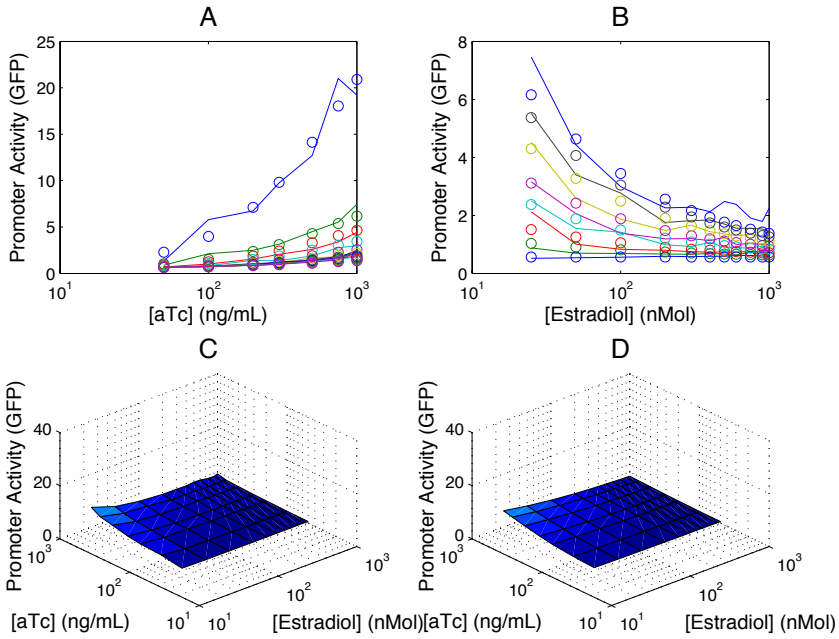

Figure 2. Script to Run Non-Linear Regression of Two-Dimensional Surface Plots. The script performs two-dimensional fitting of a function z=f(x,y) to a user define fitting function. This user defined fitting function can be defined from “ModeltoFit”. The equation takes the form of ff=f(a,x,y) where “a” is an array representing the free fitting parameters. The input arguments to the program are: 1- “a”, an array representing the starting values of the free fitting parameters (similar to that when “nlinfit” is used). The length of a must be equal to the number of parameters used in the user defined function to be fitted “ModeltoFit”. 2- “x”, a one-dimensional array representing an independent variable. 3- “y”, a one-dimensional array representing an independent variable. 4- “z”, the dependent variable matrix whose elements are functions of the mesh grid values of x and y.
The outputs of the script are: 1- The vales of the best-fitted parameters. 2- The fitted function matrix zzz 3- plots of the original data of z (lines) with fitted values (dots) plotted against x and y independently. The script overcomes the inability of the standard "nlinfit" tool in Matlab to perform two-dimensional fits. This carried out by converting the matrix form of z to a one dimensional vector zz, merging x and y into one vector xx, filling the extra needed elements of

xx as compared to zz with dummy numbers 999999 to make xx and zz suitable to be handled by “nlinfit”. After performing the fits, the script restructures the fitted values into the matrix form.

Design 2 bifurcation analysis

1D Bifurcation

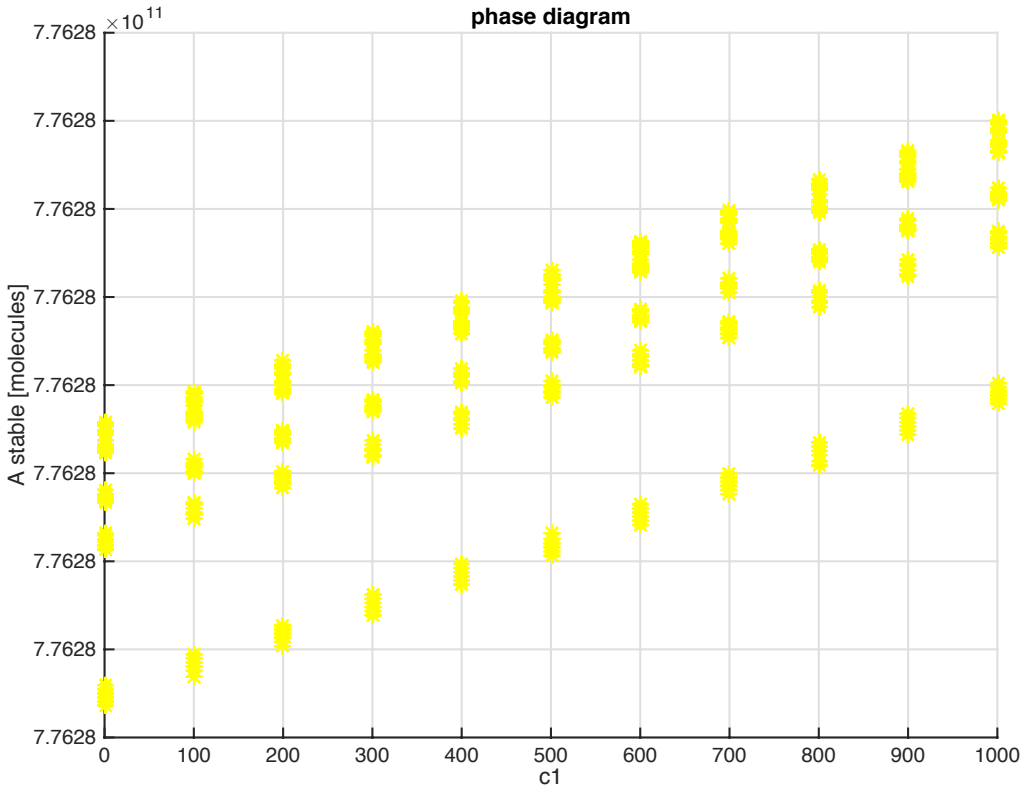

Figure 3. Bifurcation diagram. Plot of the last 100 points of each of the conditions (3 sets of equations: one multiplicative and two additives). It tests the stability of the system at a parameter (k), the system runs 100 simulations with various starting concentrations of GEV (i) and rtTa (j). All the simulations end points are plotted against the concentrations of the drugs and show the presence of the stable points. Ideally it should show two or three stable points. The parameter values were calculated based off flow cytometer data analysis.

2D Bifurcation

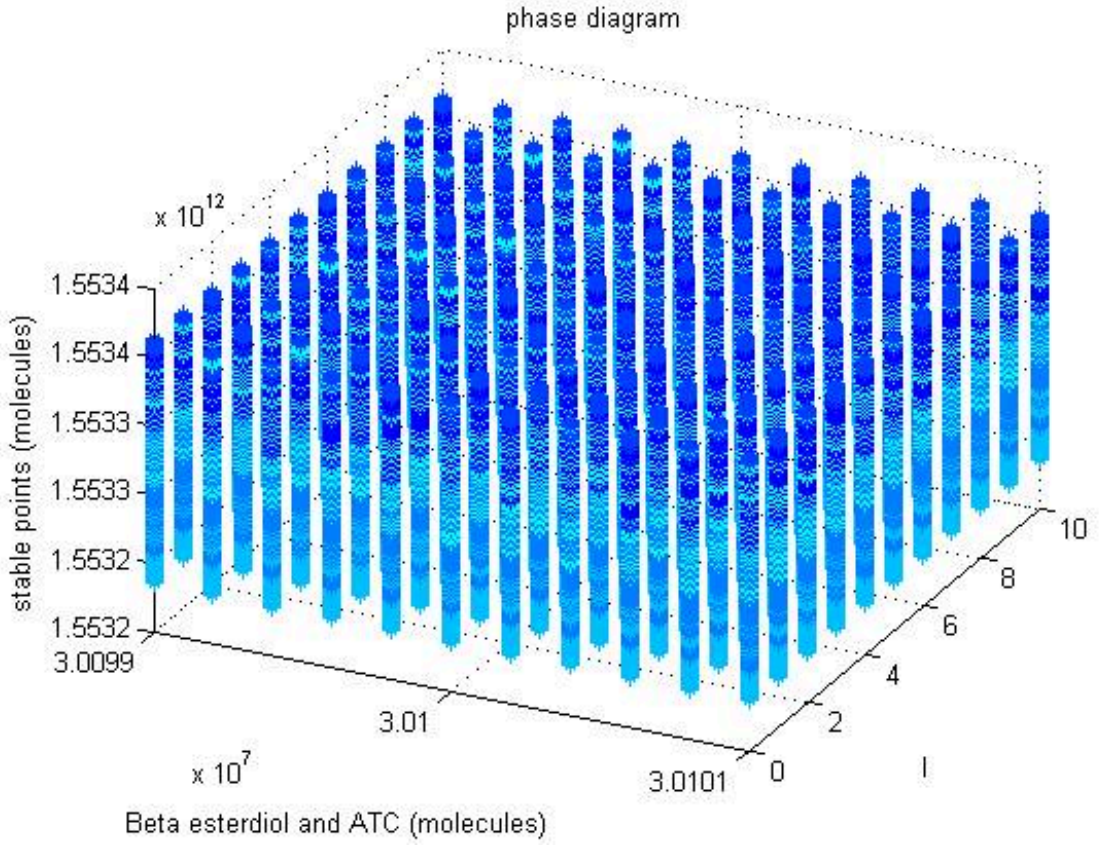

Figure 4. Bifurcation diagram. Plot of the last 100 points of each of the conditions (3 sets of equations: one multiplicative and two additives). It tests the stability of the system at two parameters (k and l), the system runs 100 simulations with various starting concentrations of GEV (i) and rtTa (j). All the simulations end points are plotted against the concentrations of the drugs and show the presence of the stable points. Ideally it should show two or three stable points. The parameter values were calculated based off flow cytometer data analysis.